DESCRIPTION

Routing Bandwidth-Reserved Connections in Information Networks

TECHNICAL FIELD

The present invention concerns the routing function in information networks, e.g. switch-based computer networks. In such a network it is necessary to determine paths from source nodes to destination nodes. This invention enhances and expands the known Dijkstra routing method to support additional types of service, e.g. reserved bandwidth service, which are not possible with the Dijkstra method. The invented method will also be called "widest-path method" throughout this description. A specific path metric is used, called "bottleneck metric" in the sequel, which was found to be compatible with the algebraic rules that govern the routing method. With this metric, it is possible to reflect realistically enough at least the bandwidth characteristics of the paths, but other characteristics may also be represented. The widest-path method can be used e.g. in connection-oriented networks as Asynchronous Transmission Mode (ATM) or Internet Stream Protocol Version II (ST.II) networks, where the routing decisions are taken at connection setup, but it is not limited thereto. It can be used to precompute paths from any source to any destination and prestore all paths until a respective one is used for a connection request. Such precomputed routing trees are advantageous in source routing methods, where the local source node tree is used to produce a source vector, which describes the path as a sequence of nodes to be covered during packet transmission. The present invention is especially useful in link-state routing mechanisms for networks, but it could be used in the

context of any routing problem for which the widest-path method is applicable and for which the bottleneck metric is an appropriate representation of the respective path characteristic, even if the context is far away from electronic network technology. As examples, passenger or goods transportation with capacity, financial, legal, or any other bottlenecks, or electronic road guide systems shall be mentioned.

BACKGROUND OF THE INVENTION

Link-state algorithms such as Open Shortest Path First (OSPF) are in common use for providing the routing function in computer networks implementing a connectionless network layer. In such cases, the network routing algorithm builds routing tables as a background task. Information about links is maintained and updated by a topology function replicated in all nodes; as a result, every node owns an image of the network, see e.g. EP 0 348 327 or EP 0 447 725. This image is used with a shortest-path algorithm to compute routes to all destinations. The routing tables, produced by the routing algorithm, normally are used to forward individual packets. With the traditional metrics, optimal paths are "shortest" paths. They are obtained by using the conventional Dijkstra method with a path "length" given by the sum of the "lengths" of the separate links contributing to the path. In such a setting, the "length" of a link is most often not its true geometrical length, but can be a value representing any characteristic of that link. In the following, "weight" will be used as the general term for such values. It could represent e.g. monetary costs for the use of that link, and one goal of the routing algorithm would be to minimize the cost of the network, while maintaining proper connectivity. It could also represent delays on that link, the goal would be to minimize the delays in network data flow. A few examples of metrics in connection with bandwidth or occupancy characteristics can be found in EP 0 276 754 and in US 4 905 233. In EP 0 276 754, a link weight approximately proportional to the occupied capacity is described and used in the Dijkstra method.

A metric that reflects the allocatable capacity available on links is also known from US 5 088 032 and US 5 067 127. In US 5 067 127, a congestion avoidance control method for communication networks is described, which uses a link weight inversely proportional to the available bandwidth and the path weight is the sum of the link weights. In US 5 088 032 a modified Ford path computation algorithm is described. There, the weight of a link can be inversely proportional to the available bandwidth, and the path weight is determined as the maximum of the weights of its links. Whereas it is stated there that other methods of finding the route with minimum metric may also be used, it is not clear at all that any other method is compatible with the metric proposed. A distance vector method is described; the Dijkstra method is not mentioned at all. As said above, the traditional Dijkstra method uses a path weight, which is determined as the sum of the weights of its links, and therefore it is no substitute for the modified Ford algorithm. Further and in contrast to the distance vector method, the widest-path method (as the Dijkstra method) builds a complete spanning tree of paths from a source to all destinations using a topology database of all nodes, their directly attached links and related link weights. This is especially useful in link-state routing mechanisms and source routing.

In virtual circuit networks, routing is connection-oriented and the routing decision is taken at connection setup. If, in addition, connections must have guaranteed bandwidth, e.g. for loss-sensitive communication, a virtual circuit network with bandwidth reservation is necessary. Examples are networks of ST.II routers and ATM networks. There, all packets or cells belonging to a connection follow the same path. In such cases, the routing algorithm applies to the routing of connection setup messages, this is also referred to "call routing".

It is a general object of this invention to avoid the different drawbacks of the prior art and to extend and modify the Dijkstra routing method in a way which allows to determine from the weights of the bottleneck link or links of

each path the "best" path, which is defined to include the "widest" bottleneck, that is the link with the most favorable (smallest or biggest) weight. It is another object to provide a link-state routing method, especially for virtual circuit networks, with guaranteed bandwidth or bandwidth reservation or with other characteristics which necessitate a bottleneck metric. A further object is to improve a network node by implementing in it a routing function enhancement comprising the widest-path method; improvements to the topology function are proposed to include in its update method a modified dampening method and/or a bandwidth encoding method to enable consideration of dynamically varying available bandwidths. Further disclosed is a network comprising improved nodes which may be mixed with normal nodes not supporting the devised enhancement.

SUMMARY OF THE INVENTION

The above objects are accomplished by enhancing and extending the Dijkstra routing method by applying an appropriate metric to determine link weights and path weights. An appropriate metric must reflect at least approximately the characteristics of the paths to be taken into account in the routing method and it must be compatible with this method. As was found, the bottleneck metrics comply with these constraints. They include metrics which are defined so that the weight of a path is given by the maximum of the weights of its links, and a link or path with smaller weight is the better link or path, respectively. In this case, with the widest-path method, the best paths are still paths with minimal weight in this case, as with the Dijkstra method. A formal description of such an example of the method in algorithm form is given in the appendix. There, a case is selected where the operation of link weight summation in the Dijkstra method is always replaced by a maximum operation which has the maximum of the link weights as its result. This definition means that the weight of a concatenated path is now the maximum of the weights of its links instead of

the sum. It is possible to formally proof that the algebraic rules which govern the method hold for both operations. The beauty of the widest-path method is that it is easy to implement and can replace the Dijkstra method, where appropriate, without complications. Clearly, the bottleneck metrics include other metrics, too. As examples, the minimum (or another extremum) of the component link weights (or their absolute values) can be used to determine the path weight directly or after further calculation, provided that the calculation applied is a non-decreasing function. The median of component link weights or the component link weight closest to a predetermined target value can be used, if these reflect the path characteristic to be described. As a rule, an operation on the weights of the component links of a path is applied to select at least one link (the "bottleneck link") of the path, and the path's weight is then determined from the weights of its bottleneck links.

In the context of communication network routing, the metric reflects the allocatable capacity available on links and the widest-path method is used for the computation of the path with the highest allocatable capacity. In link-state routing, network nodes share link state information that reflects the available bandwidth on each of the links of the network. This is performed by encoding the available capacities as link weights and using a known distribution mechanism, called "topology function", for transmission. As the available capacity varies very dynamically, it is necessary to prevent excessive amounts of link state updates. This is known as "dampening" and an appropriate dampening method is described. The routing function can be applied to connection setup requests instead of individual packets. The widest-path method computes paths from any source to any destination, using the information obtained from the topology function. The paths can be stored and used to route connection requests as they arrive. One feature of the "widest-path" definition is that either a connection setup can be routed along a widest path, no matter how much bandwidth it requests, or it cannot be routed at all in the network. In other words, the method guarantees that the connection will find a path with sufficient bandwidth, assuming there

FIG. 3 shows an exponential bandwidth encoding format for link-state update information.

FIG. 4 illustrates call routing and the related information flow.

DETAILED DESCRIPTION OF AN EMBODIMENT ACCORDING TO THE INVENTION

A path is the concatenation of links, also called "component links" of the path, between network nodes. The width Cpath of a path is defined as the minimum of the available capacity on each of the component links. The available capacity is the bandwidth, in bits per second, that can be allocated to user connections. Therefore, the capacity bottleneck link determines what capacity is available on a path. A "widest path" is a path that, among all paths between one source and one destination, has the largest width.

Figure 1 illustrates a widest-path example in a domain including nodes 1 to 7 of an arbitrarily meshed network. Links of different available bandwidths are shown and the respective bandwidth is depicted by the width of the link connecting line. As is shown, a widest path from node 1 to 2 is the path 1-4-5-2, with an assumed width of say 40 Mb/s, determined by link 5-2. Whereas link 1-3 (100 Mb/s) is broader than 1-4 (60 Mb/s), path 1-3-2 is narrower than 1-4-5-2. It has a width of only 20 Mb/s, say, due to the bottleneck link 3-2. Weights are applied to the links in such a way that a link with smaller weight is not narrower than a link with bigger weight. Then, the widest link is a link with smallest weight and the narrowest link is a link with biggest weight. As an example, the weight Wlink of a link is defined as

Wlink = Cmax - Clink,

where Cmax is a constant assumed to be larger than any link capacity (say Cmax = 16 Gb/s). Clink is the available capacity, or bandwidth of the link.

equal-weight routes. In the example, path 1-4-5-2 precedes path 1-4-7-5-2 of equal weight which is determined by bottleneck link 5-2 in both cases.

To make the method work in a link-state, connection-oriented routing environment, the nodes of the network need new capabilities. Figure 1B shows a network node according to the invention including a known topology function 10. A widest-path generator 12 is connected to the topology function 10. Upon connection requests 13 from a network user, the widest path is assigned to route the connection. Further, link-state update information 14 is exchanged between network nodes to keep the topology function up to date. A bandwidth information update module 11 is connected to the topology function to include bandwidth information in the link-state update information 14 for variable available link capacity. Module 11 is comprising an encoder to format a bandwidth information to be sent out by the node, a receiver for receiving and, if necessary, decoding bandwidth information of other nodes, and a dampening mechanism avoiding immediate updating reaction to small bandwidth changes.

Module 11 encodes the available bandwidth Clink on a link as a 16-bit weight, see Figure 3. This format is used for compatibility reason with existing link-state algorithms. An exponential notation is used in order to cover a range from 1 bit/s to Cmax = 16 Gb/s. The encoding uses 8 as the exponentiation basis, 3 bits of exponent 21 starting from the most significant bit 23, and 13 bits of mantissa 22, ending with the least significant bit 24. There may be several ways to encode a specific capacity Clink. Among all encodings (exp, mant) for one capacity Clink, only the one with the smallest exponent is declared valid. This rule allows to put away with decoding capacities before manipulating them, because the usual comparisons on "long integers" apply. Namely, if c, c' are the 16-bit encodings of link capacities C, C', then

$$C < C' \iff c < c' \iff W > W'.$$

Changing available bandwidth of a link with immediate bandwidth updating of all nodes, which is similar to changing its weight, can lead to disastrous scenarios, such as storms of link-state updates propagating through the network during a period of very frequent connection setups. This can lead to congestion, excess transient loops and similar problems often encountered in situations of overcorrections. To avoid this, a dampening method was defined which only invokes link-state updates for a link when a significant change appears, e.g. when an amount of its bandwidth has been reserved which is larger than a certain dampening threshold. For example, five connections for a fraction of Mbits/s each on a link of several Gbits/s occurring in a second would lead to five times distributing a change of not even 0.1% of the link's capacity, and probably to recomputation of the topology through all nodes. This is clearly unacceptable. The dampening method is based on the fraction of link bandwidth reserved. To achieve this goal, a threshold MaxDBandwidth must be provided that during the change of the dynamic link weight decides whether the new link advertisement should be started or not. Because of this requirement, every link must, beside the bandwidth weight field, contain a cumulated, not flushed, change in weight called delta-bandwidth. Every connection setup or release changes the delta-bandwidth and checks whether it exceeds the threshold. If it does, new topology update is propagated. One problem still remains, namely the "opaqueness" of the delta-bandwidth cost to all nodes. When the bandwidth of a link has been changed and "absorbed" by the delta-bandwidth field, it can potentially not be advertised for a long period of time. A possible routing mismatch during this period of time could happen, although this is rather unlikely, because the threshold should be so small that not distributing the delta should be negligible for routing. Nevertheless, a periodic timer for each node link is introduced, which is started whenever delta-bandwidth is changed from 0 to a value not equal to 0 and reset each time delta-bandwidth is set to 0. When the timer expires, it flushes delta-bandwidth if necessary. The dampening constant of 5% of the available link bandwidth is based on the behavior of a typical scenario assumed with either uniform or exponential size distribution of the requests

arriving at a constant rate with a maximum size of 10% of the link bandwidth.

Most of the up-to-date link-state routing protocols offer the capability of dividing the routing domain into subdomains. A topology information is summarized at the boundaries of the subdomains and only the summary is distributed. Certain constraints have nevertheless to be met to guarantee the non-ambiguity of the distributed information. The method of widest-path areas is proposed which allows to intermix subdomains understanding widest-path and standard metrics with those only understanding standard metrics. An example of such a mixed network is depicted in Figure 2. Three widest-path areas 16,17,18 of different topologies are shown imbedded in a network with areas 19,20 of standard nodes. On boundaries 15 of two subdomains with different characteristics, the unsupported metrics are simply dropped. This allows a gradual introduction of the widest-path method in routing domains. Here, the necessary changes for a OSPF standard routing protocol are described to get so called WET-OSPF, but other mixed networks are possible. WET denotes the three option bit names W, E, and T, of which only W is related to the widest-path area method. E and T are not relevant here.

In this context, network nodes are called "routers". Widest-path areas consist only of routers supporting the widest-path method. This is determined by a similar mechanism as the one used to have all routers in a stub area agree about the stub property. A new option bit is introduced, called W-bit. Routers of a widest-path area set this bit sending so-called hello-packets on area interfaces and refuse to build adjacency to routers in the area that do not have this bit set. Interfaces of widest-path routers connecting to a standard area will not have this bit set in the hello-packets, but only in the options field of the link advertisement for summary links, so that distribution of bandwidth metrics over the border of two widest-path areas will work. Moreover, a new time constant WET-MinLSInterval is introduced, beside the MinLSInterval of OSPF. The MinLSInterval is used

APPENDIX

1. Formalism and Assumptions

- uses $\exists$ operator to check for existence
- $Head(), Tail()$ return head or tail of a list. 0 if empty
- $Head+(e,q), Tail+(e,q)$ adds a element e to list q only if it is not yet in the list
- $Head-(q), Tail-(q)$ remove head. tail of list and returns removed value or 0 if list empty
- $Insert(e.k,q)$ inserts element e into list q at position k
- MAX gives the maximum of its arguments
- [illegible] gives the number of elements in a list
- $[x]$ is the element at place x in a array or list or set
- node 1 is the source
- R is number of nodes
- { } denotates a empty set or list

2. Algorithm (continued)

```
WHILE c≠0 DO
    FOR i:=1 TO #Node[c].Links /* number of this node's links */
        dst := Node[Node[c].Links[i].destination];
        Head+(c, Spf);
        IF (∃ x | Node[dst].Links[x].destination = c) AND dst ∉ SPF
            /* checks whether a back link exists and destination not
               already computed on tree ?? */
            dist := MAX ( Node[c].Links[i].cost, Length(Route[c]) );
            IF (dist < Length(Route[dst]) OR Length(Route[dst])=0)
                Route[dst]:= Route[c];
                Tail+(Element(c, i), Route[dst]);
                k:=1;
                WHILE k < #Cand AND Length(Route[k]) < dist
                    k:= k+1;
                    END;
                Insert(dst, k, Cand ); /* insert sorted on candidate list */
                END;
            END;
        END;
    c= Head-( Cand );
    END;
```

CLAIMS

1. Method for determining the best path of a plurality of paths from a source to a destination in or through a network of nodes and links between the nodes, wherein
   - each link has assigned a link weight reflecting a selected link characteristic,
   - the path weight of a, preferably each, concatenated path is determined by the link weights of its components,
   - the path weights of said plurality of paths determine said best path,
   - a best path tree is constructed from said source to at least one destination using a topology database containing the nodes, their attached links, and the related link weights for each concatenated path taken into account,
   - a subset is selected containing at least one link from the set of component links of said path by applying an operation on the link weights of said path's component links, and
   - a path weight is determined of said path from the link weights of its selected links.

2. The method according to claim 1, wherein an extremum of the component link weights is used for the subset selecting step.

3. The method according to claim 1, wherein the maximum of the component link weights defines the weight of a concatenated path.

4. The method according to claim 1, wherein, in a digital communication network, the link weights reflect available bandwidth on the links, all link weights are positive, and smaller link weight means broader bandwidth.

5. The method according to one or more of claims 1 to 3, wherein the link weights reflect transport capacity, in particular in a road network or for goods or passengers.

6. Routing device (10,12), for a network of nodes (1-9) and links between the nodes, comprising
   - a memory storing information about link states including weights reflecting a characteristic of the links, and
   - a best path generator determining the weight of a path from the link weights of its component links, and determining the best path using the path weights of a plurality of paths from a source to a destination in or through the network,

   wherein
   - said memory contains the nodes, their attached links, and the related link weights,
   - said best path generator (12) comprises selection means for selecting a subset of at least one link from the set of component links of a path by applying an operation on the weights of the path's component links, and weighting means for determining the weight of a concatenated path from the link weights of its selected links.

7. The routing device according to claim 6, wherein the link weights reflect available bandwidths on the links in a digital communication network.

8. The routing device according to claim 7 for a network with dynamically changing available bandwidth, comprising a bandwidth information update module (11), including
   - an encoder which exponentially encodes the available bandwidth on a link, and
   - a dampening mechanism for avoiding immediate updating reaction to small bandwidth changes.

9. Network node (1-9), comprising a routing device (10,12) according to one or more of the claims 6 to 8.

10. Network, in particular digital communication network, comprising at least one routing device (10,12) according to one or more of the claims 6 to 8.

11. Mixed network, comprising at least one node according to claim 9 and one or more other nodes, and applying a routing protocol using the method according to one or more of the claims 1 to 4 for subdomains (15-20) of said mixed network.

12. Use of a method according to one or more of claims 1 to 4 or of a routing device according to one or more of claims 6 to 8, to enable reserved-bandwidth services in a virtual circuit network.

13. Use of a method according to one or more of claims 1 to 4 in a link-state routing protocol of or in a network using source routing.

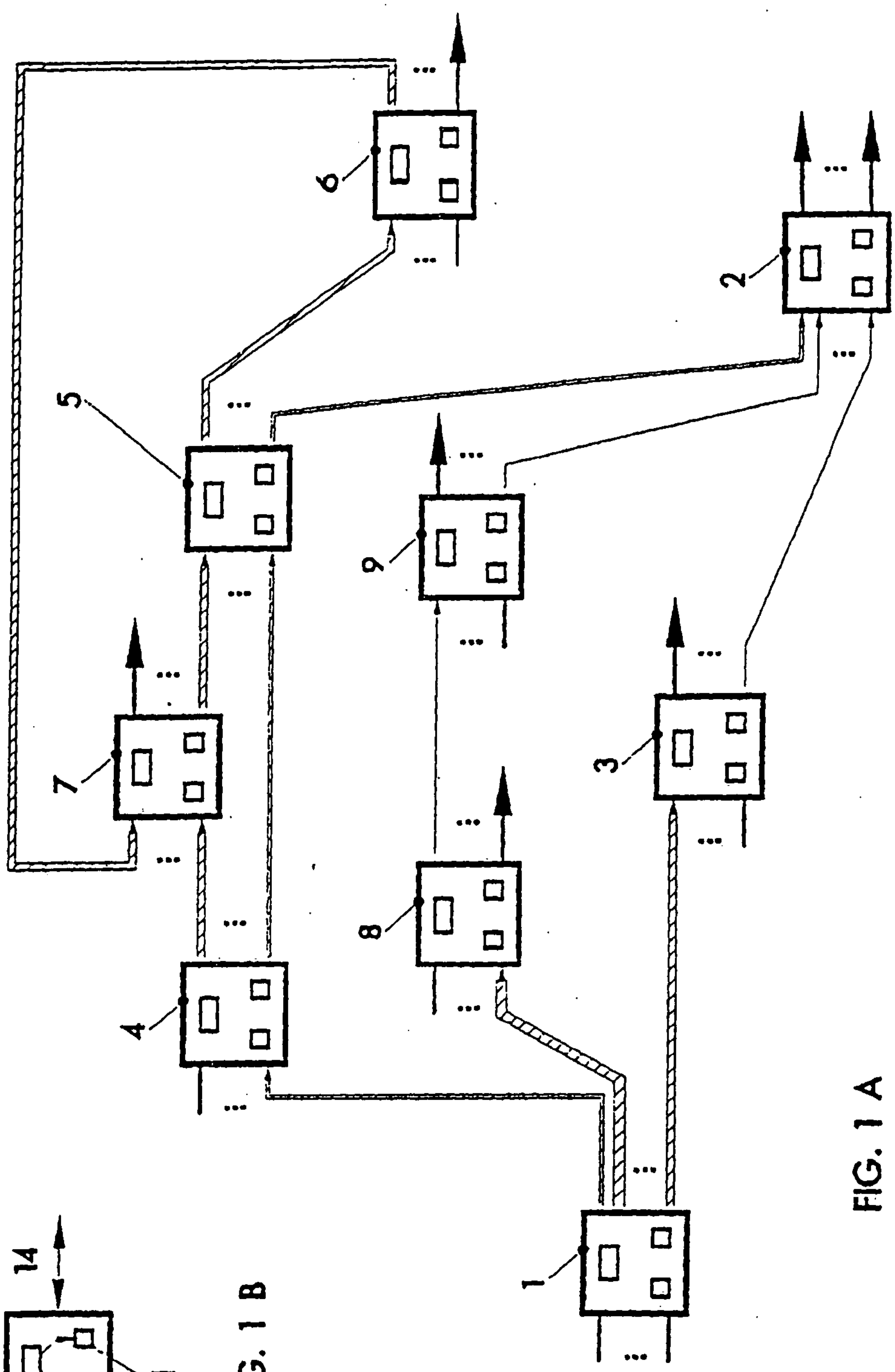

FIG. 1 A

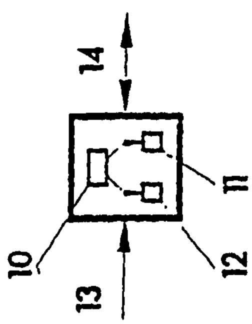

FIG. 1 B

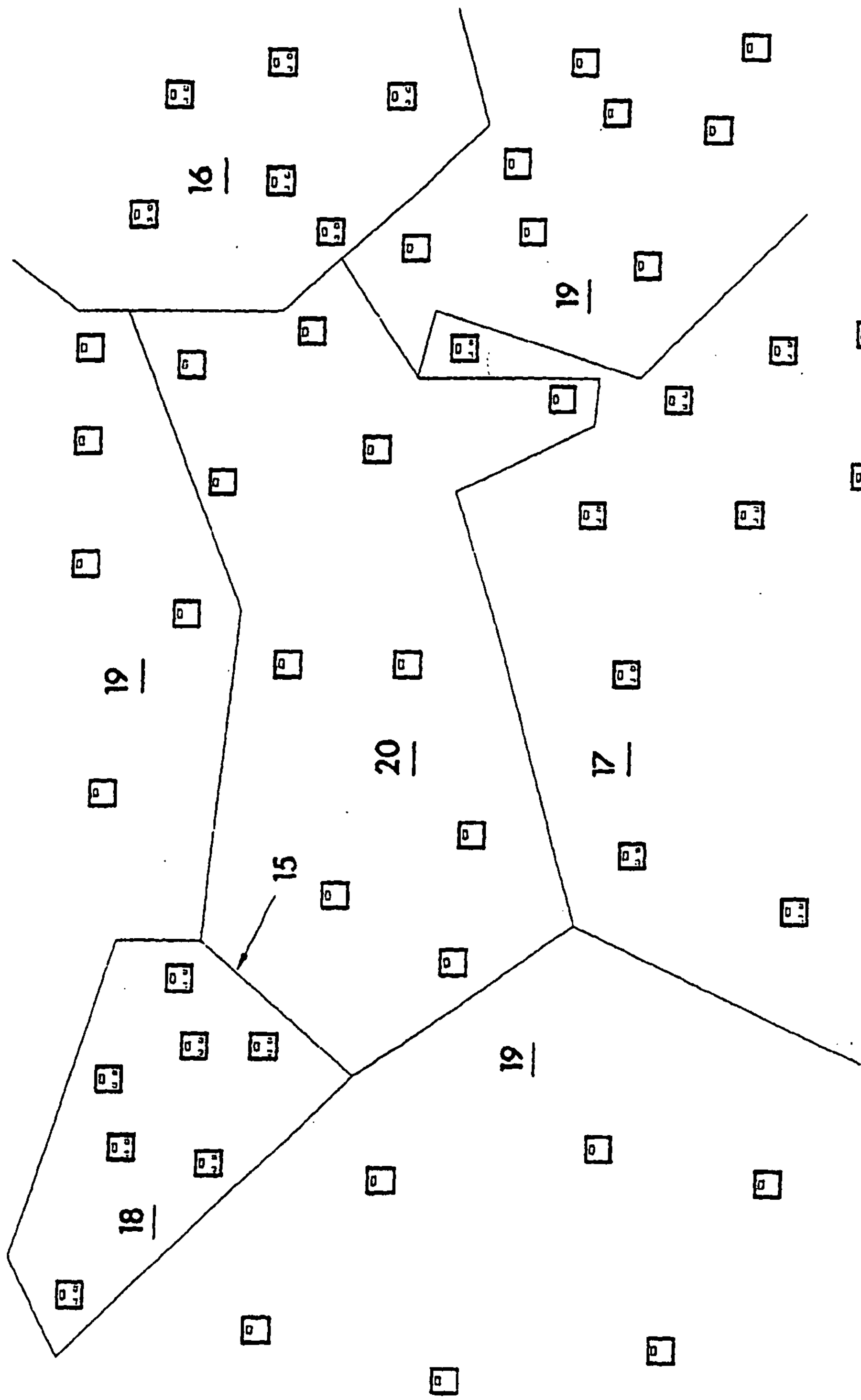

FIG. 2

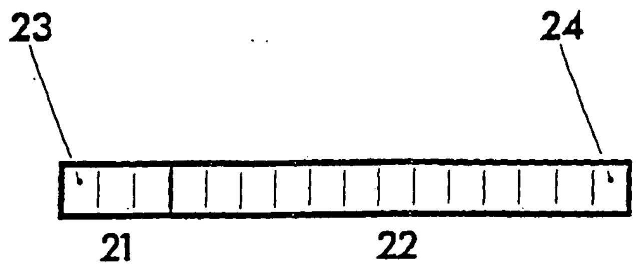

FIG. 3

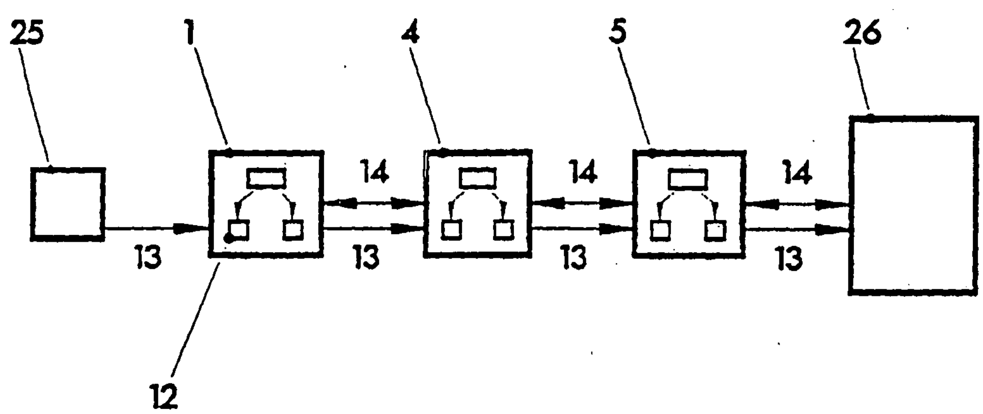

FIG. 4

INTERNATIONAL SEARCH REPORT

International Application No
PCT/EP 93/03683

C.(Continuation) DOCUMENTS CONSIDERED TO BE RELEVANT

Category *	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
	column 2, line 21 - column 7, line 45; fig. 1-3. --	
A	EP, A2, 0 447 725 (DIGITAL EQUIPMENT CORPORATION) 25 September 1991 (25.09.91), abstract; column 1, line 3 - column 3, line 55. --	1,6
A	EP, A2, 0 423 053 (INTERNATIONAL BUSINESS MACHINES) 17 April 1991 (17.04.91), abstract; column 1, line 1 - column 3, line 55. --	1,6
A	US, A, 4 905 233 (CAIN et al.) 27 February 1990 (27.02.90), abstract; column 4, line 12 - column 12, line 63; fig. 1-4B. --	1,6
A	US, A, 5 088 032 (BOSACK) 11 February 1992 (11.02.92), column 1, line 60 - column 3, line 2; column 2, line 32 - column 7, line 34; fig. 1-6. --	1-6
A	US, A, 5 067 127 (OCHIAI) 19 November 1991 (19.11.91), abstract; column 1, line 8 - column 3, line 5; column 9, line 27 - column 12, line 55; fig. 1,2,9-13. ----	1-6

ANHANG	ANNEX	ANNEXE
zum internationalen Recherchenbericht über die internationale Patentanmeldung Nr.	to the International Search Report to the International Patent Application No.	au rapport de recherche international relatif à la demande de brevet international n°

PCT/EP 93/03683 SAE 84019

In diesem Anhang sind die Mitglieder der Patentfamilien der im obengenannten internationalen Recherchenbericht angeführten Patentdokumente angegeben. Diese Angaben dienen nur zur Unterrichtung und erfolgen ohne Gewähr.	This Annex lists the patent family members relating to the patent documents cited in the above-mentioned international search report. The Office is in no way liable for these particulars which are given merely for the purpose of information.	La présente annexe indique les membres de la famille de brevets relatifs aux documents de brevets cités dans le rapport de recherche international visée ci-dessus. Les reseignements fournis sont donnés à titre indicatif et n'engagent pas la responsibilité de l'Office.

Im Recherchenbericht angeführtes Patentdokument Patent document cited in search report Document de brevet cité dans le rapport de recherche	Datum der Veröffentlichung Publication date Date de publication	Mitglied(er) der Patentfamilie Patent family member(s) Membre(s) de la famille de brevets	Datum der Veröffentlichung Publication date Date de publication
EP A2 348327	27-12-89	EP A3 348327 JP A2 2041053 US A 4873517	11-03-92 09-02-90 10-10-89
EP A1 498967	19-08-92	AU A1 10873/92 AU B2 649101 CA AA 2069195 CN A 1066948 JP A2 5063726 MX A1 9200497 US A 5265091	20-08-92 12-05-94 14-08-92 09-12-92 12-03-93 02-03-93 23-11-93
EP A1 276754	03-08-88	AT E 86817 DE CO 3878941 DE T2 3878941 EP B1 276754 ES T3 2039482 FR A1 2610161 FR B1 2610161 US A 4831649	15-03-93 15-04-93 17-06-93 10-03-93 01-10-93 29-07-88 25-03-94 16-05-89
EP A2 447725	25-09-91	AU A1 69248/91 AU B2 619701 CA AA 2035231 EP A3 447725 JP A2 4223632 US A 5128926	03-10-91 30-01-92 22-09-91 16-09-92 13-08-92 07-07-92
EP A2 423053	17-04-91	JP A2 3139936 US A 5321815	14-06-91 14-06-94
US A 4905233	27-02-90	keine - none - rien	
US A 5088032	11-02-92	keine - none - rien	
US A 5067127	19-11-91	CA AA 2025846 JP A2 3108845	22-03-91 09-05-91